The Multi-Level Creation Process in Flow Network Reliability Estimation

Héctor Cancela, Leslie Murray, Gerardo Rubino

The Multi–Level Creation Process in Flow Network Reliability Estimation

Héctor Cancela [1] Leslie Murray [2] Gerardo Rubino (**Speaker**) [3]

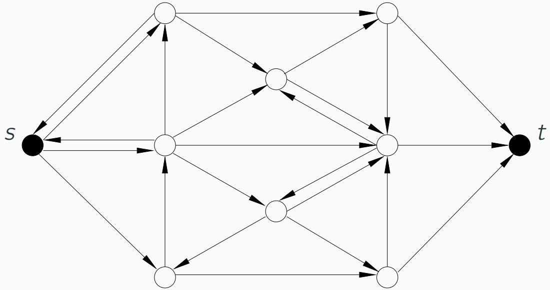
[1]Facultad de Ingeniería, UDELAR, Montevideo, Uruguay

[2]Facultad de Ingeniería, UNR, Rosario, Argentina

[3]INRIA Rennes, France

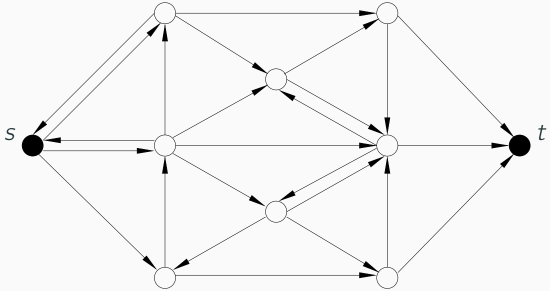
July 6, 2018

Introduction: Flow Network



$$\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X}) \begin{cases} \mathcal{V} : \text{set of } n \text{ nodes} \\ \mathcal{E} : \text{set of } m \text{ links} \\ \mathbf{X} : \text{capacity per link, } (X_1, \ldots, X_m) \end{cases}$$

$M(\mathbf{X})$ = maximum amount of flow generated in s that can reach t



- The capacities X_i, $i = 1, \ldots, m$, change due to failures
- $\mathbf{X} = (X_1, \ldots, X_m)$, is a random vector in $\boldsymbol{\Omega} = (\Omega_1, \ldots, \Omega_m)$
- $\zeta = \mathbb{P}\{M(\mathbf{X}) < T\}$. If the network is highly reliable, $\zeta \ll 1$

Aim of this talk

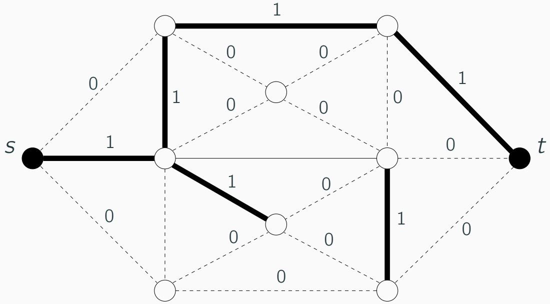
To introduce an efficient method for the estimation of ζ, when $\zeta \ll 1$

Talk Outline

1. Network Static Model (Review)
    - Basic Setting / Crude Monte Carlo
    - The Creation Process
    - Splitting on the the Creation Process
2. Stochastic Flow Network Model
    - Multi–Level Creation Process
    - Splitting on the Multi–Level Creation Process
3. Experimental Results
    - Efficiency Analysis
4. Conclusions and Lines of Future Work

Network Static Model

$$\mathbf{X} = (X_1, X_2, \cdots, X_m) \to X_i = \begin{cases} 1 \text{ w.p. } r_i & i^{th} \text{ link operative} \\ 0 \text{ w.p. } q_i = 1 - r_i & i^{th} \text{ link failed} \end{cases}$$



$$\phi(\mathbf{X}) = \begin{cases} 1 & \text{if } \ s \text{ and } t \textit{ are not connected} \\ 0 & \text{if } \ s \text{ and } t \textit{ are connected} \end{cases} \to \ \zeta = \mathbb{P}\{\phi(\mathbf{X}) = 0\}$$

Crude Monte Carlo

1. Sample N independent instances of $\mathbf{X}$: $\mathbf{X}^{(i)}, \; i = 1, \cdots, N$
2. Compute $\phi(\mathbf{X}^{(i)}), \; i = 1, \cdots, N$
3. Estimate ζ as:

$$\widehat{\zeta} = \frac{1}{N} \sum_{i=1}^{N} \phi(\mathbf{X}^{(i)})$$

$$\text{RE} \triangleq \frac{\sqrt{\mathbb{V}\{\widehat{\zeta}\}}}{\mathbb{E}\{\widehat{\zeta}\}} = \frac{\sqrt{\zeta(1-\zeta)}}{\sqrt{N}\,\zeta} \underset{\text{if } \zeta \ll 1}{\overset{\uparrow}{\approx}} \frac{1}{\sqrt{N\,\zeta}} \quad \to \infty \quad \text{if } \zeta \to 0 \text{ with } N \text{ fixed}$$

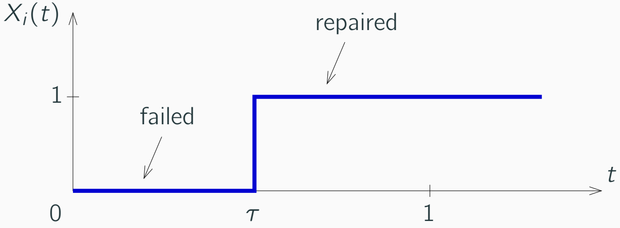
Crude MC does not solve the problem...

Rare event problem.

The Creation Process

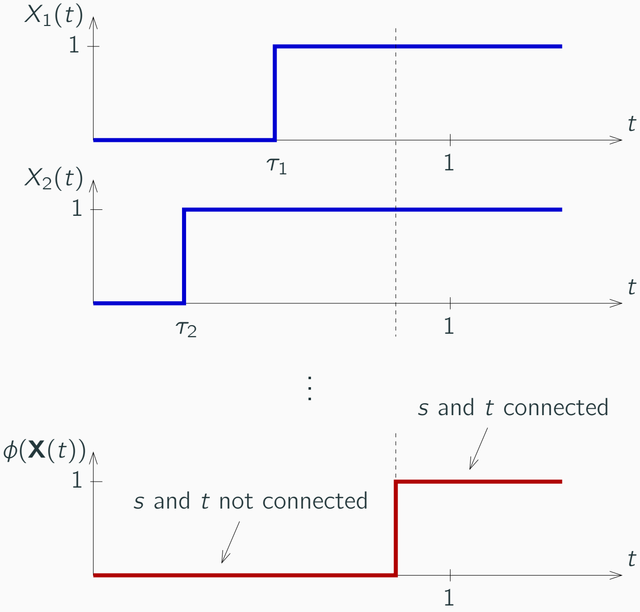
Time enters: $\mathbf{X} \to \mathbf{X}(t) = (X_1(t), X_2(t), \cdots, X_m(t))$

For each component, and for the whole system,



Basic Idea

If links are repaired in a time proportional to their own single reliabilities, s and t become connected in a time proportional to the network's reliability


$X_1(t)$
1
t
τ_1
1
$X_2(t)$
1
t
τ_2
1
⋮
s and t connected
$\phi(\mathbf{X}(t))$
1
s and t not connected
t
1

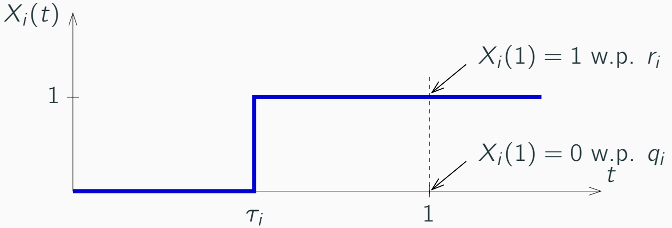
Let τ_i be the repair time of component i, with

- $\tau_i \sim \exp(\lambda_i),\ i = 1, \ldots, m$
- $\lambda_i = -\ln(q_i)$

Then:

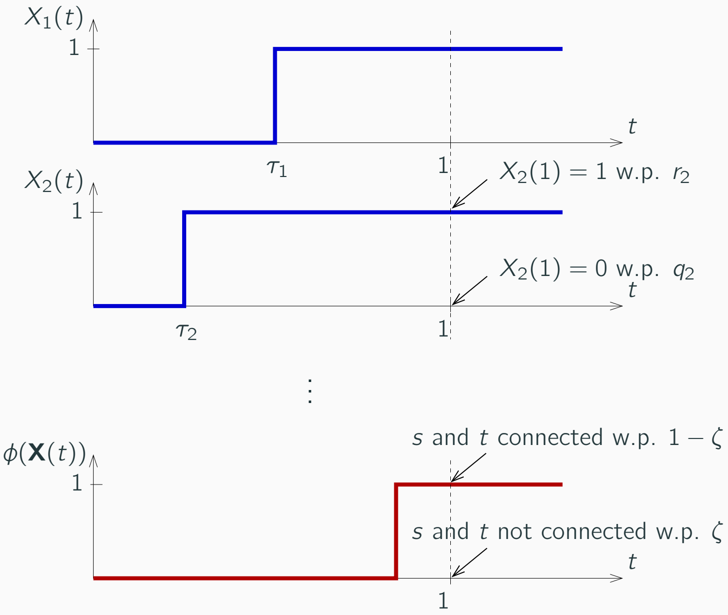
$$\mathbb{P}\{X_i(1) = 1\} = \mathbb{P}\{\tau_i \leq 1\} = 1 - e^{-\lambda_i} = 1 - e^{\ln(q_i)} = r_i$$

$$\mathbb{P}\{X_i(1) = 0\} = \mathbb{P}\{\tau_i > 1\} = e^{-\lambda_i} = e^{\ln(q_i)} = q_i$$



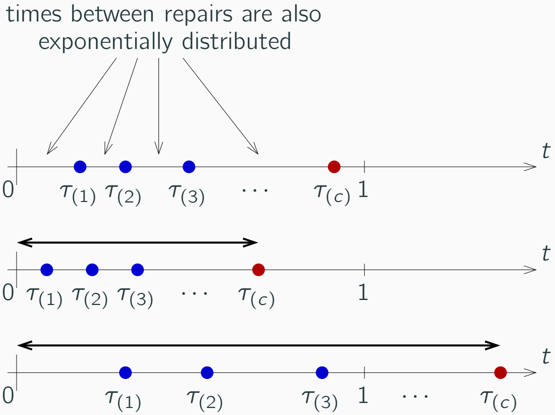
Finally, if $\mathbb{P}\{X_i(1) = 0\} = q_i,\ i = 1, \ldots, m,$ then $\mathbb{P}\{\phi(\mathbf{X}(1) = 0)\} = \zeta$

The Creation Process



The Creation Process

Let τ_c be the repair time of the system. and consider the following point process.



Algorithm

Sample sequences $\{\tau_{(1)}, \tau_{(2)} \cdots, \tau_{(c)}\}$, see them as trajectories in a one–dimensional space and reduce the problem to: $\zeta = \mathbb{P}\{\tau_c > 1\}$

The Creation Process / Crude MC

1. Sample N times the sequence $\{\tau_{(1)}, \tau_{(2)} \cdots, \tau_{(c)}\}$
2. From each one of them, read the value of the random variable $\tau_{(c)}$ and compute the function:

$$I = \begin{cases} 1 & \text{if } \tau_{(c)} > 1, \\ 0 & \text{otherwise} \end{cases} \quad \rightarrow \quad I^{(i)}, \;\; i = 1, \cdots, N$$

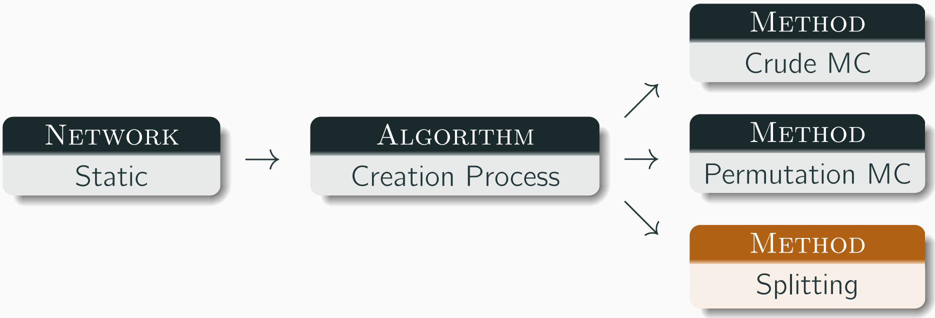
3. Estimate ζ as:

$$\widehat{\zeta} = \frac{1}{N} \sum_{i=1}^{N} I^{(i)}$$

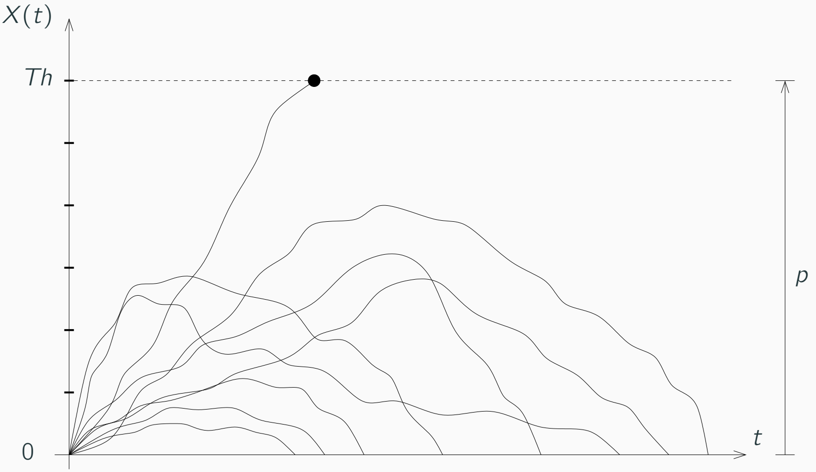
Still the same Crude Monte Carlo estimation...

It has to be improved by means of a variance reduction method

Methods based on the Creation Process

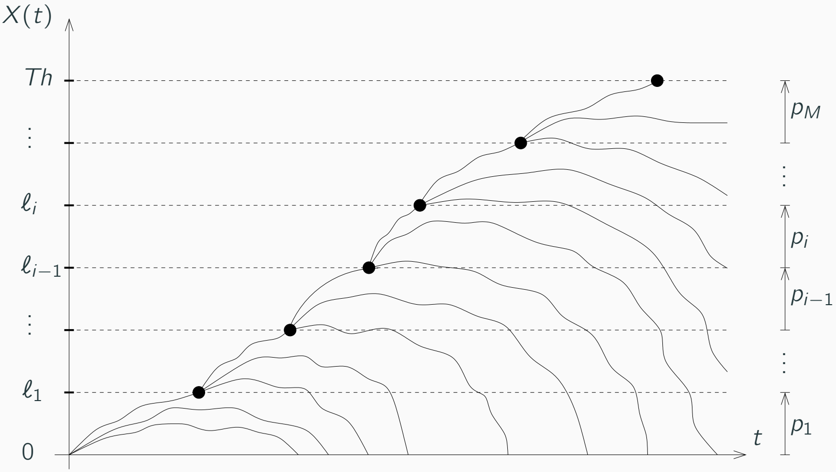


Splitting



$$\zeta = p$$

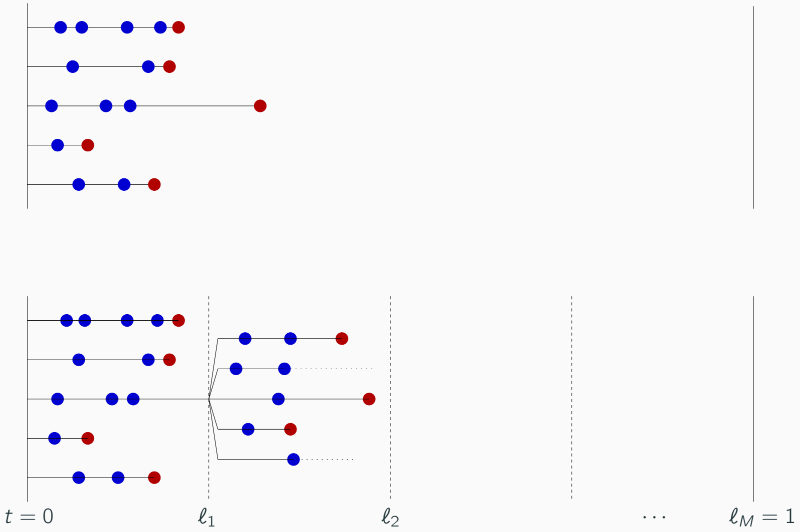
Splitting



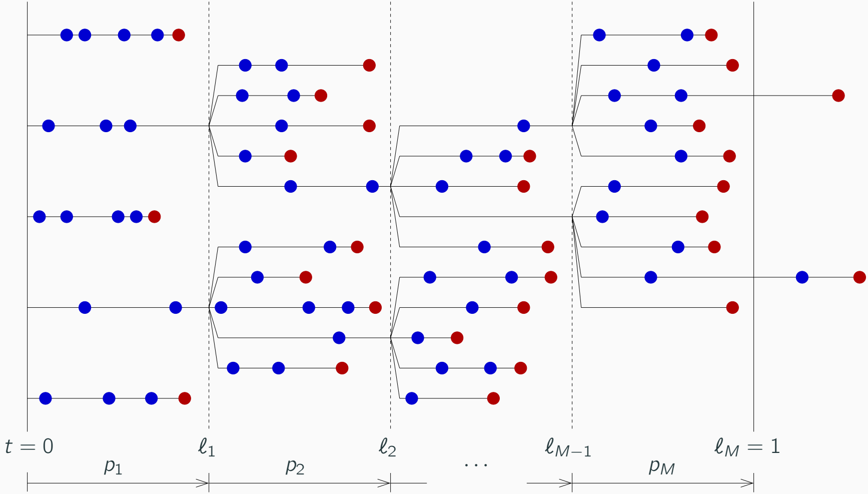
$$\zeta = \prod_{i=1}^{M} p_i \quad \rightarrow \quad \widehat{\zeta} = \prod_{i=1}^{M} \widehat{p}_i$$

Splitting on the Creation Process

Splitting on the Creation Process



Splitting on the Creation Process



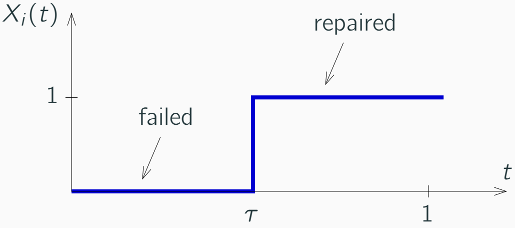
$$\zeta = \prod_{i=1}^{M} p_i \quad \rightarrow \quad \widehat{\zeta} = \prod_{i=1}^{M} \widehat{p}_i$$

Network: Static → **Algorithm**: Creation Process →
- **Method**: Crude MC
- **Method**: Permutation MC
- **Method**: Splitting

Network: Stochastic Flow → **Algorithm**: Multi–Level CP →
- **Method**: Crude MC
- **Method**: Splitting

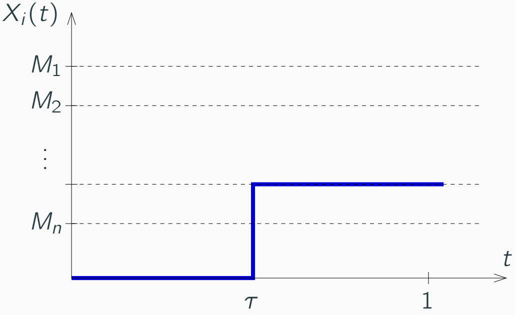
Static Network Model:

$$X_i = \begin{cases} 1 & \text{w.p. } r_i, \\ 0 & \text{w.p. } q_i \end{cases}$$

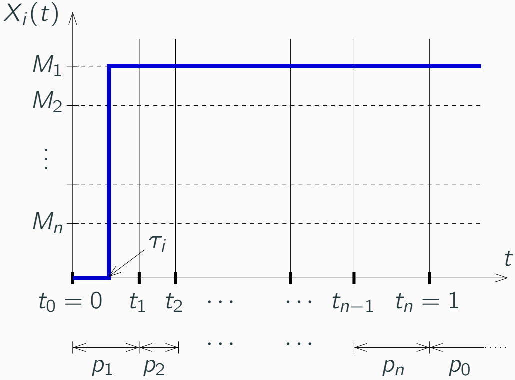


Stochastic Flow Network:

$$X_i = \begin{cases} M_1 & \text{w.p. } p_1, \\ M_2 & \text{w.p. } p_2, \\ & \vdots \\ M_n & \text{w.p. } p_n, \\ 0 & \text{w.p. } p_0. \end{cases}$$

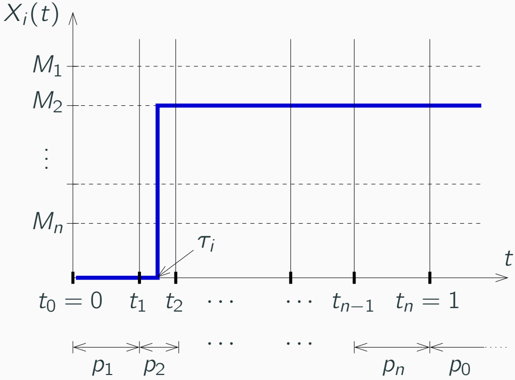


Multi–Level CP

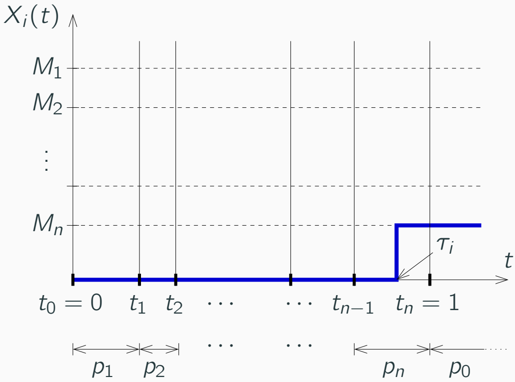


$$
\begin{array}{lrl}
\text{If} & 0 < \tau_i \le t_1 & \text{set } X_i(t) = M_1 \text{ and keep this value forever,} \\
\text{If} & t_1 < \tau_i \le t_2 & \text{set } X_i(t) = M_2 \text{ and keep this value forever,} \\
 & \vdots & \\
\text{If} & t_{n-1} < \tau_i \le t_n & \text{set } X_i(t) = M_n \text{ and keep this value forever,} \\
\text{If} & 1 < \tau_i & \text{leave } X_i(t) = 0 \text{ forever.}
\end{array}
$$

Multi–Level CP

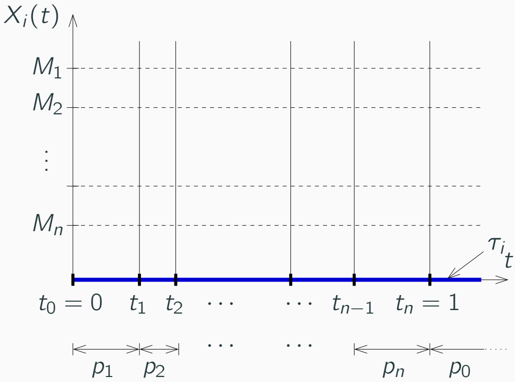


$$
\begin{array}{lrl}
\text{If} & 0 < \tau_i \le t_1 & \text{set } X_i(t) = M_1 \text{ and keep this value forever,} \\
\text{If} & t_1 < \tau_i \le t_2 & \text{set } X_i(t) = M_2 \text{ and keep this value forever,} \\
 & \vdots & \\
\text{If} & t_{n-1} < \tau_i \le t_n & \text{set } X_i(t) = M_n \text{ and keep this value forever,} \\
\text{If} & 1 < \tau_i & \text{leave } X_i(t) = 0 \text{ forever.}
\end{array}
$$



If	$0 < \tau_i \le t_1$	set $X_i(t) = M_1$ and keep this value forever,
If	$t_1 < \tau_i \le t_2$	set $X_i(t) = M_2$ and keep this value forever,
	$\vdots$	
If	$t_{n-1} < \tau_i \le t_n$	set $X_i(t) = M_n$ and keep this value forever,
If	$1 < \tau_i$	leave $X_i(t) = 0$ forever.

Multi–Level CP



$$
\begin{array}{lll}
\text{If} & 0 < \tau_i \le t_1 & \text{set } X_i(t) = M_1 \text{ and keep this value forever,} \\
\text{If} & t_1 < \tau_i \le t_2 & \text{set } X_i(t) = M_2 \text{ and keep this value forever,} \\
 & \vdots & \\
\text{If} & t_{n-1} < \tau_i \le t_n & \text{set } X_i(t) = M_n \text{ and keep this value forever,} \\
\text{If} & 1 < \tau_i & \text{leave } X_i(t) = 0 \text{ forever.}
\end{array}
$$

As τ is exponentially distributed:

$$f_\tau(t) = \lambda e^{-\lambda t} \text{ and } F_\tau(t) = 1 - e^{-\lambda t}.$$

Two important issues related to τ are not yet defined:

❶ Which is the rate, λ, of its distribution?

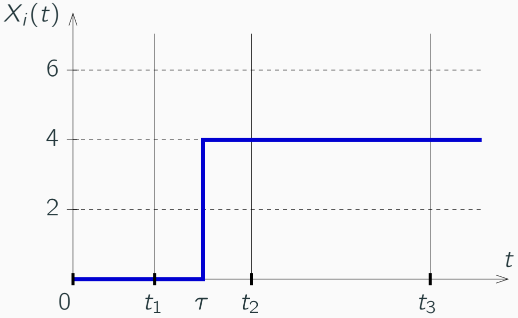
$$\mathbb{P}\{\tau > 1\} = 1 - F_\tau(1) = 1 - (1 - e^{-\lambda}) = e^{-\lambda} = p_0$$

$$\text{Then: } \lambda = -\ln(p_0)$$

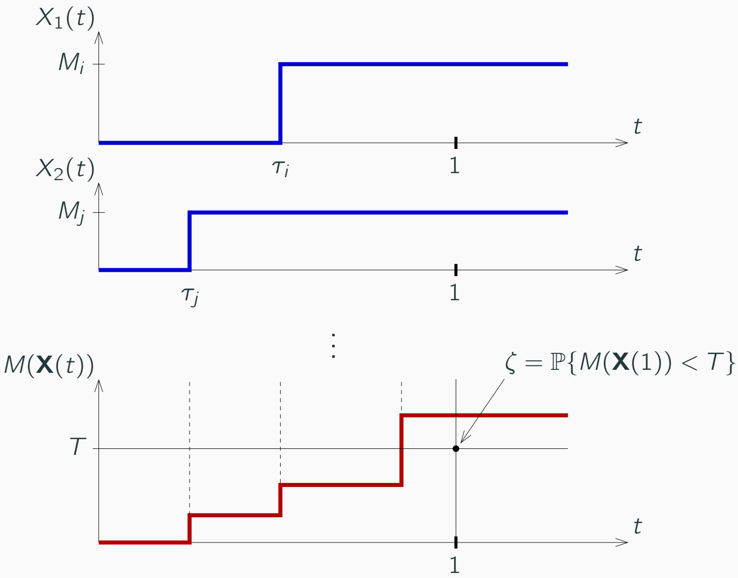
❷ Which are the values of the times $t_1, t_2, \cdots, t_{n-1}$?

$$\mathbb{P}\{t_{i-1} < \tau \leq t_i\} = p_i \;\rightarrow\; t_i = \frac{\ln(1 - p_1 - \cdots - p_i)}{\ln(p_0)}$$

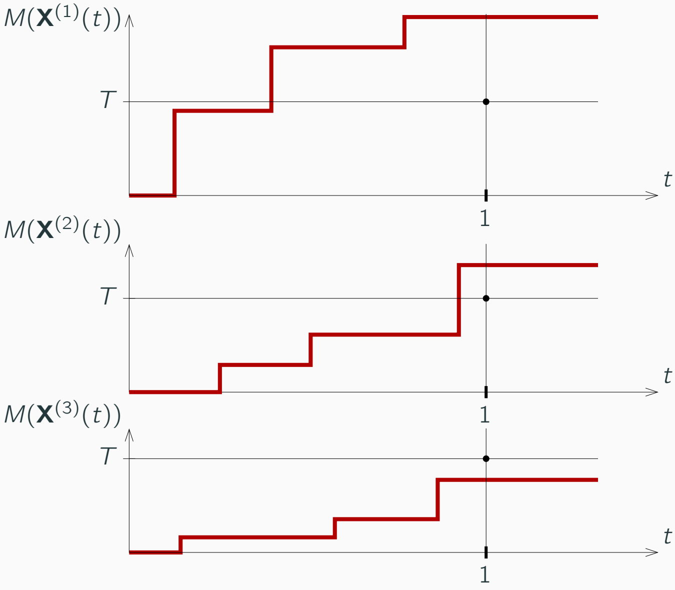
$$X_i = \begin{cases} 0 & \text{w.p. } 0.05 \\ 6 & \text{w.p. } 0.48 \\ 4 & \text{w.p. } 0.29 \\ 2 & \text{w.p. } 0.18 \end{cases} \quad \rightarrow \quad \begin{aligned} \lambda &= 2.996 \\ t_1 &= 0.218 \\ t_2 &= 0.491 \\ t_3 &= 1.000 \end{aligned}$$



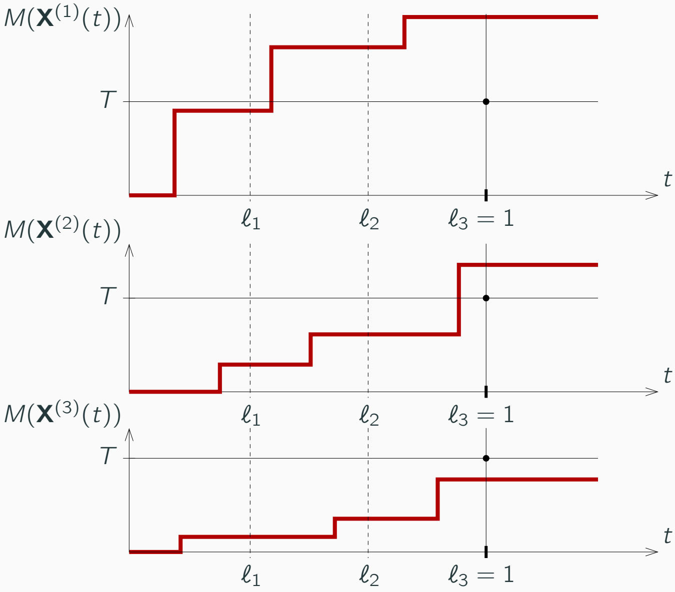
Splitting on the Multi–Level CP



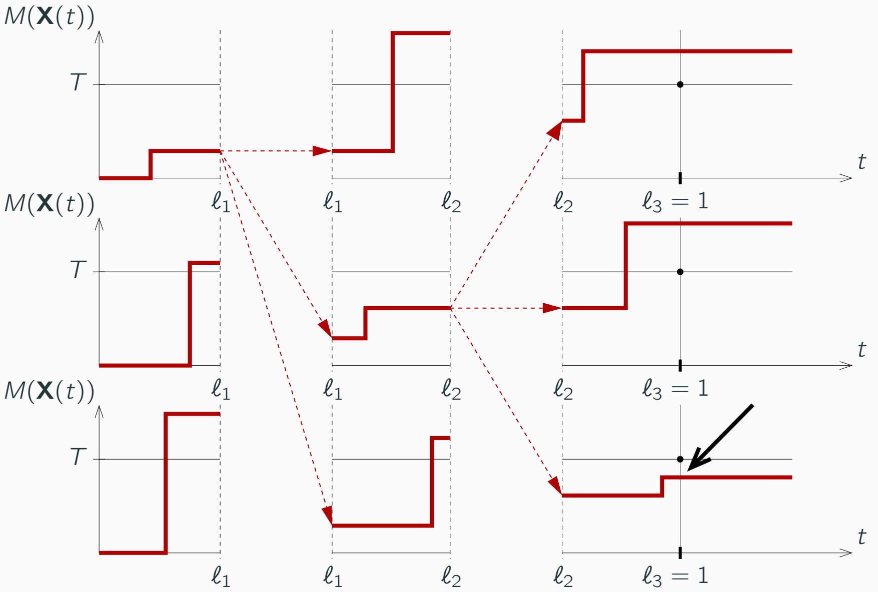
Splitting on the Multi–Level CP



Splitting on the Multi–Level CP



Splitting on the Multi–Level CP



Multi–Level CP / Continuous Capacities

So far, links' capacities were modelled in terms of a discrete distribution,

$$X_i = \begin{cases} M_1 & \text{w.p. } p_1, \\ M_2 & \text{w.p. } p_2, \\ & \vdots \\ M_n & \text{w.p. } p_n, \\ 0 & \text{w.p. } p_0. \end{cases}$$

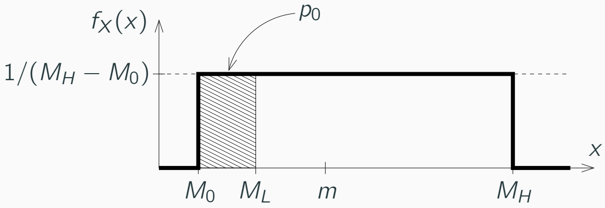
but the model can be changed by another one in which the non–zero capacities follow a continuous distribution like, for example:

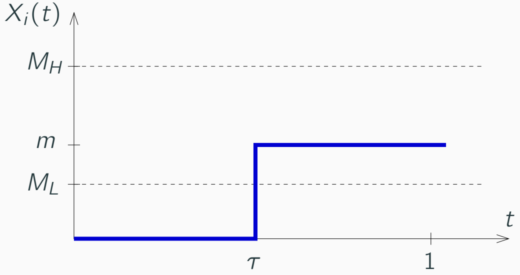
- $X_i = 0$ with probability p_0
- X_i uniformly distributed between M_L and M_H, whenever $X_i \neq 0$

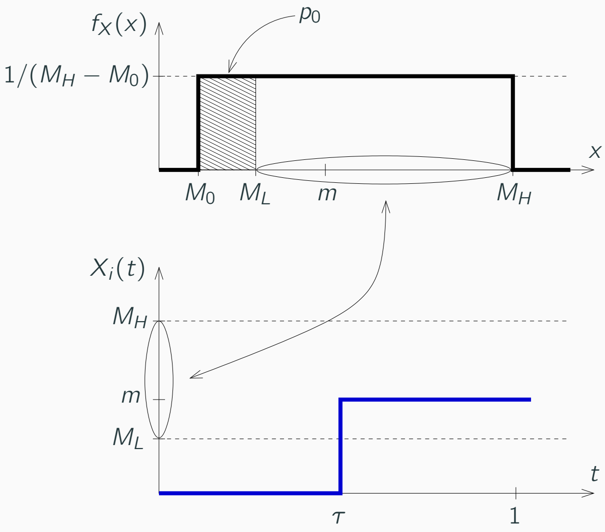
$$X_i = B_i \times U_i$$

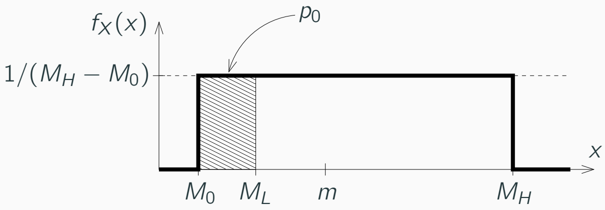
where $B_i \sim \text{Ber}(1 - p_0)$ and $U_i \sim \text{Unif}(M_L, M_H)$

Multi–Level CP / Continuous Capacities






$f_X(x)$
p_0
$1/(M_H - M_0)$
x
M_0
M_L
m
M_H
$X_i(t)$
M_H
m
M_L
t
τ
1



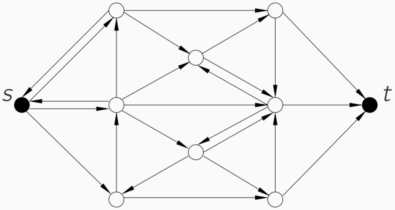
Time τ is still exponentially distributed and,

1. The rate, λ, of its distribution is:

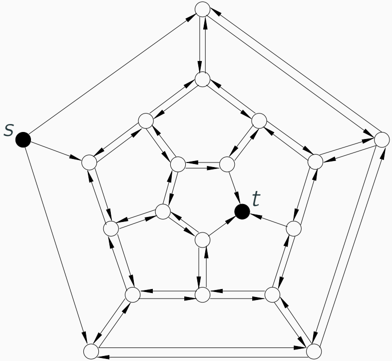
$$\lambda = -\ln(p_0) = -\ln\left(\frac{M_L - M_0}{M_H - M_0}\right)$$

2. The capacity, m, for any sampled time, τ, is:

$$m = M_0 + (M_H - M_0)\, e^{-\lambda\tau}$$



Fishman Network



Dodecahedron

Efficiency Analysis I

Network: *Fishman* Network
Number of Replications: 10^2 (effort per level $= 10^3$)
Links' Capacities: $X_i = 0$ w.p. p_0 and, while not zero, Unif$(100, 200)$
Results: $\widehat{\zeta}$, RE and *Speedup*

	$T = 300$	$T = 250$	$T = 200$	$T = 150$	$T = 100$
$p_0 = 0.1$	3.18E−01	1.39E−01	6.32E−02	3.54E−02	2.58E−03
$p_0 = 0.01$	2.99E−02	8.18E−03	5.85E−04	3.02E−04	1.92E−06
$p_0 = 0.001$	3.01E−03	7.56E−04	5.88E−06	3.00E−06	1.94E−09
$p_0 = 0.0001$	3.02E−04	7.39E−05	5.89E−08	2.99E−08	1.94E−12
$p_0 = 0.1$	0.42%	0.63%	0.80%	0.80%	1.07%
$p_0 = 0.01$	0.84%	1.30%	1.52%	1.40%	2.25%
$p_0 = 0.001$	1.14%	1.34%	1.73%	1.70%	2.26%
$p_0 = 0.0001$	1.18%	1.74%	1.85%	1.79%	2.57%
$p_0 = 0.1$	0.5	0.5	0.8	1.3	8.6
$p_0 = 0.01$	1.4	2.2	15.4	37.4	2,317.6
$p_0 = 0.001$	5.4	13.3	936.3	1,702.9	1,412,607.4
$p_0 = 0.0001$	42.7	72.4	57,604.2	112,470.0	851,348,214.3

Efficiency Analysis II

Network: Dodecahedron
Number of Replications: 10^2 (effort per level $= 10^3$)
Links' Capacities: $X_i = 0$ w.p. p_0 and, while not zero, Unif$(100, 200)$
Results: $\widehat{\zeta}$, RE and *Speedup*

	$T = 300$	$T = 250$	$T = 200$	$T = 150$	$T = 100$
$p_0 = 0.1$	3.52E−01	1.60E−01	7.18E−02	4.13E−02	2.93E−03
$p_0 = 0.01$	2.98E−02	8.27E−03	5.96E−04	3.06E−04	2.05E−06
$p_0 = 0.001$	3.00E−03	7.57E−04	6.12E−06	3.01E−06	1.97E−09
$p_0 = 0.0001$	3.02E−04	7.52E−05	5.90E−08	2.98E−08	1.98E−12
$p_0 = 0.1$	0.33%	0.60%	0.75%	0.83%	1.06%
$p_0 = 00.1$	0.94%	1.31%	1.28%	1.43%	2.43%
$p_0 = 000.1$	1.16%	1.53%	1.83%	1.92%	2.21%
$p_0 = 0000.1$	1.27%	1.83%	1.87%	1.89%	2.67%
$p_0 = 0.1$	0.6	0.5	0.8	1.0	9.1
$p_0 = 0.01$	1.1	2.2	22.4	33.1	2,014.2
$p_0 = 0.001$	5.3	11.0	756.6	1,369.4	1,523,596.1
$p_0 = 0.0001$	32.9	66.3	55,301.5	107,402.7	635,779,816.5

Comparison to other papers' results

Network: Dodecahedron
Number of Replications: 6×10^6 (*)
Links' Capacities: $X_i = 0$ w.p. q and, $X_i = 1$ w.p. $1 - q$

Permutation Monte Carlo

q	10^{-6}	10^{-5}	10^{-4}	10^{-3}	10^{-2}	0.10	0.15
$\widehat{\zeta}$	2.99E−06	3.02E−05	3.00E−04	3.00E−03	3.06E−02	3.57E−01	5.54E−01
RE	2.08%	1.94%	1.53%	2.11%	1.09%	0.31%	0.28%

Splitting on the Multi–Level CP

q	10^{-6}	10^{-5}	10^{-4}	10^{-3}	10^{-2}	0.10	0.15
$\widehat{\zeta}$	3.01E−06	3.00E−05	2.99E−04	3.00E−03	3.07E−02	3.58E−01	5.55E−01
RE	0.76%	0.62%	0.48%	0.38%	0.22%	0.08%	0.05%

(*) The number of times each method applies the Ford Fulkerson algorithm.

Conclusions

- Multi–Level CP is a generalization of the classical Creation Process.
- Multi–Level CP performs as the base for a Splitting application (Permutation algorithms are not possible).
- Splitting on the Multi–Level CP is a new proposal, reason why, more insightful analysis (variance, error, robustness, . . .) are yet to be done.
- Splitting on the Multi–Level CP performs very efficiently when networks are highly reliable. Experimental results are very encouraging.

Incremental Ford Fulkerson

- Every time a threshold ℓ_i is crossed, the condition $M(\mathbf{X}(\ell_i)) < T$ is checked by means of the Ford Fulkerson algorithm.
- Ford Fulkerson algorithm proceeds iteratively, *flooding* the links up to the maximum possible flow from s to t.$^{(*)}$
- At the crossing of threshold ℓ_1 links are *empty* and the are *flooded* up to an amount $\mathbf{L}_1 = (L_1, \ldots, L_m)$ of flow.
- At the crossing of threshold ℓ_2 (for all the created trajectories) flows are incremented, starting all of them from $\mathbf{L}_1$.

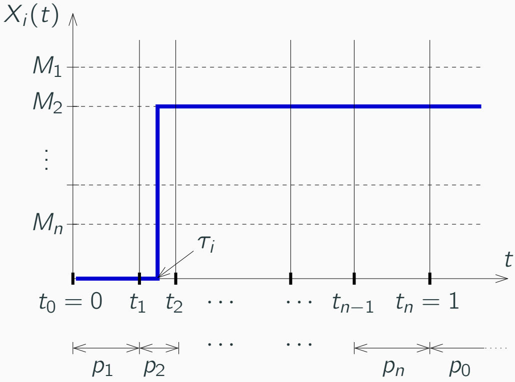
Algorithm Improvement

- In standard Ford Fulkerson flows increase, always starting from zero, over empty links.
- In *Incremental Ford Fulkerson*, at every Splitting stage, flows increase starting from the values reached in the previous stage.

Incremental Ford Fulkerson significantly reduces the simulation time.

$^{(*)}$It is an imaginary flow, only for the purposes of the Ford Fulkerson algorithm.

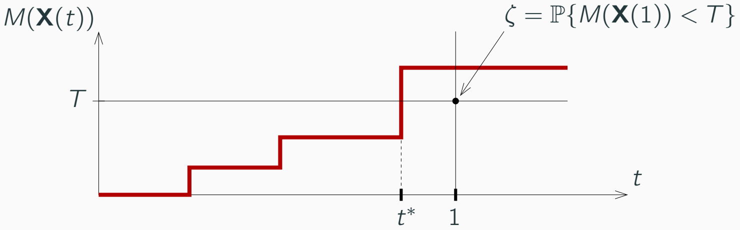
Capacity Sampling Plan



- Largest *capacity* M_1 comes from the lowest times, $(0, t_1]$.
- Second largest *capacity* M_2 comes from times $(t_1, t_2]$.

"*Capacities* decrease as the corresponding times τ_i increase"

But this rule is not an operating requirement, it is optional...



- For highly reliable networks most trajectories exceed T in $t^* < 1$.
- For all such trajectories there is some work to be done and certain amount of splitting to perform.
- In order to reduce this work and, therefore, the simulation time, it is convenient that these trajectories exceed T *as soon as possible*.
- To achieve (or at least to get close to) this objective, larger capacities should be sampled earlier.